

MP4

Problem Description:

Credit card numbers follow certain patterns. A credit card number must have between 13 and 16 digits. It must start with:

4 for Visa cards

5 for Master cards

37 for American Express cards

6 for Discover cards

In 1954, Hans Luhn of IBM proposed an algorithm for validating credit card numbers. The algorithm is useful to determine if a card number is entered correctly or if a credit card is scanned correctly by a scanner. Almost all credit card numbers are generated following this validity check, commonly known as the Luhn check or the Mod 10 check, which can be described as follows (for illustration, consider the card number 4388576018402626):

Step 1 Double every second digit from right to left. If doubling of a digit results in a two-digit number, add up the two digits to get a single-digit number.

$$2 * 2 = 4$$

$$2 * 2 = 4$$

$$4 * 2 = 8$$

$$1 * 2 = 2$$

$$6 * 2 = 12 (1 + 2 = 3)$$

$$5 * 2 = 10 (1 + 0 = 1)$$

$$8 * 2 = 16 (1 + 6 = 7)$$

$$4 * 2 = 8$$

Step 2 Now add all single-digit numbers from Step 1.

$$4 + 4 + 8 + 2 + 3 + 1 + 7 + 8 = 37$$

Write method which implements steps 1 and step 2:

```
public static int doubleDigitsAndSumSingleDigits( String creditCard );
```

Step 3 Add all digits in the odd places from right to left in the card number.

$$6 + 6 + 0 + 8 + 0 + 7 + 8 + 3 = 38$$

Write a method which implements step3:

```
public static int addOddNumbersFromRightToLeft( String creditCard );
```

Step 4

Sum the results from Steps 1, 2 and Step 3.

$$37 + 38 = 75$$

Write a method which implements step 4:

```
public static int sum ( int step1andSep2, int step3 );
```

Step 5

If the result from Step 4 is divisible by 10, the card number is valid; otherwise, it is invalid. For example, the number 4388576018402626 is invalid, but the number 4388576018410707 is valid

```
public static boolean isValid ( String CreditCard );
```

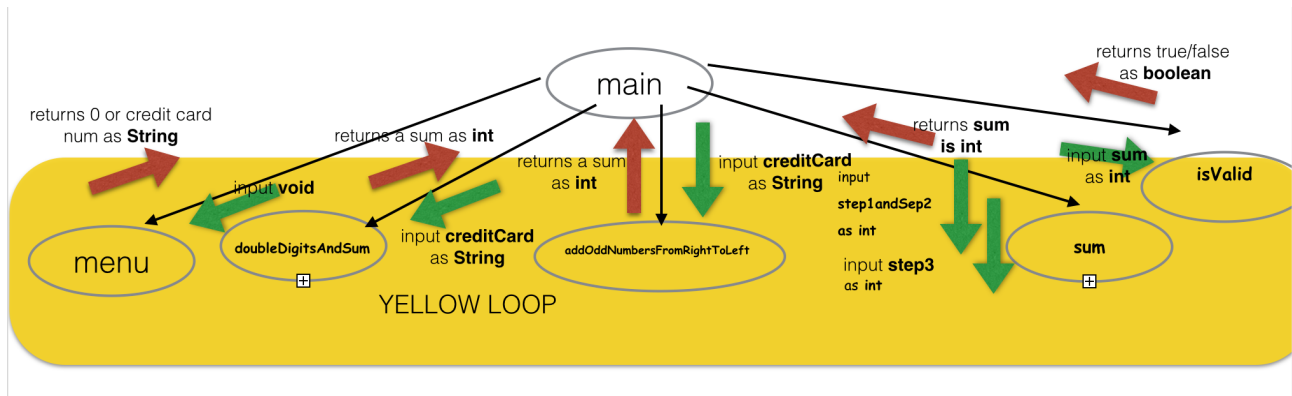
Write a program that prompts the user to enter Q/q to quit or C/c to type a credit card number. Display whether the number is valid or invalid every time the user enters a credit card number.

Your main

method calls a menu() from within a while loop which exits when the user quits.

Method menu(): Displays Q/q or C/c. If the user enters C, it read the credit number and returns it as String. If the user types Q/q returns 0.

Your main calls a method menu. Then, applies the Luhn's algorithm by calling the methods created in steps 1 to 5 as shown below:



Sample 1:

Enter a credit card number as a long integer: 4246345689049834

4246345689049834 is invalid

Sample 2:

Enter a credit card number as a long integer: 4388576018410707

4388576018410707 is valid